CANmunicator User Manual

QuickCAN interface program

Revision 1.3

Date: 11-2005

1. Introduction ---------- 1

2. Quick Start ---------- 1

3. Disclaimer ---------- 1

4. System requirements ---------- 1

5. Specification ---------- 2

6. QuickCAN hardware pin-out and LED indication ---------- 2

7. Device driver installation ---------- 3

8. CANmunicator ---------- 6

8.1. Clear All ---------- 6

8.2. Get Message ---------- 7

8.3. Enable Scan ---------- 7

8.4. Save ---------- 7

8.5. Add message ---------- 7

8.6. Send ---------- 8

8.7. Arm Trigger ---------- 8

8.8. Abort Tx ws Enable Tx ---------- 8

8.9. Filter Registers ---------- 8

8.10. Baud rate Register ---------- 9

8.11. Register map ---------- 9

8.12. MCP2515 Status ---------- 10

8.13. Reset MCP2515 ---------- 11

8.14. Exit ---------- 11

8.15. Operation mode ---------- 11

8.16. Baud Rate ---------- 11

9. USB-2-CAN Communication protocol ---------- 12

9.1. Command R ---------- 12

9.2. Command W ---------- 13

9.3. Command C: ---------- 13

9.4. Command X ---------- 14

9.5. Command I ---------- 15

9.6. Command A: ---------- 15

9.7. Command T: ---------- 16

9.8. Command M: ---------- 16

9.9. Command D: ---------- 17

9.10. Command Q: ---------- *17*
9.11. Command V: ---------- *18*
9.12. Command Z: ---------- *18*
9.13. Command J: ---------- *19*

10. Initialization ---------- **21**

11. Revision History ---------- **23**

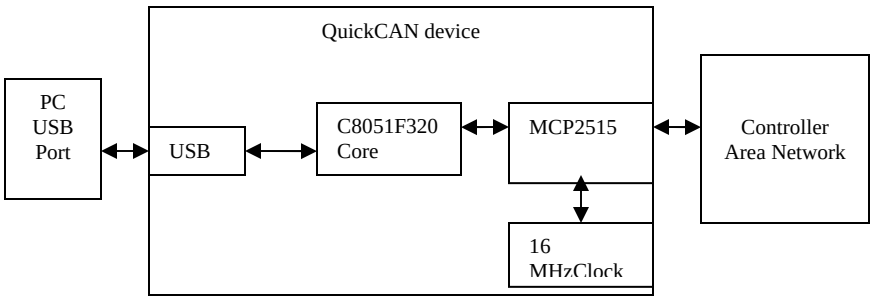
1. Introduction

QuickCAN device provides a simple, reliable and high-performance connection between your PC and the controller area network (CAN) via USB port. No external power is required as QuickCAN device is powered by built-in USB port.

QuickCAN package comes with a Windows interface application program called CANmunicator. This program initializes QuickCAN MCP2515 firmware at startup, and provides point and click access to all MCP2515 internal registers. It also provides a user-friendly interface to send and receive CAN messages that can be either standard or extended frames.

QuickCAN package also provides the USB-2-CAN communication protocol, which offers users the flexibility to develop their own applications.

The following block diagram illustrates the operation of the QuickCAN device.



2. Quick Start

2.1. Download QuickCAN Application (QuickCAN.exe) from Q-Proto system web site www.qprotos.com/download.html and install it to your local.
2.2. Connect the QuickCAN device to the PC via USB A-B cable.
2.3. Install the Windows device driver (see chapter 6).
2.4. Launch the QuickCAN.exe from Windows <start>-<QuickCAN> menu.
2.5. Ready to use.

3. Disclaimer

While every effort has been made to ensure that the information contained in this manual is accurate and complete, no liability can be accepted for any error and omission. Q-Proto system reserves the right to change the specification of the hardware and software described herein at any time prior notice.

Q-Proto System shall not be liable for direct, indirect, incidental, general or consequential damage from the use of the products from Q-Proto system. If you do not agree with these terms, do not buy the products.

4. System requirements

4.1. Windows 98SE, 2000, XP operating system
4.2. Pentium II 350MHz or above
4.3. Minimum 256M RAM
4.4. One available USB 1.1 or above port on the PC

5. Specification

5.1. High speed 8051 compatible microcontroller C8051F320 with built in USB functionality.
5.2. CAN2.0A and B support.
5.3. Microchip MCP2515 CAN controller running at 16 MHz.
5.4. Microchip MCP2551 CAN transceiver.
5.5. Self powered by USB port.
5.6. The CANmunicator initializes the MCP 2515 firmware at start up. The device is ready to send and receive CAN messages.
5.7. Documented USB to CAN communication protocol allows users to develop their own applications for extensive usage.
5.8. Full USB1.1 speed at 12MHz.
5.9. DB9 Male connector for CAN network.
5.10. Support loopback and listen operations.
5.11. Up to 1 M bit software selectable CAN baud rate.
5.12. Windows driver for Win98 SE, 2000 and XP is included.

6. QuickCAN hardware pin-out and LED indication

6.1. DB9 Male pin-out

Pin 1	RST (Not used for CAN bus)
Pin 2	CAN Low
Pin 3	GND
Pin 4	C2D (Not used for CAN bus)
Pin 5	Shield GND
Pin 6	GND
Pin 7	CAN High
Pin 8	C2CK (Not used for CAN bus)
Pin 9	V++ (optional)

Pin 1, Pin 4 and Pin 8 are used for C8051F320 programming purpose and they shall not be used for CAN connection. Misuse of these pins may result in damage of the QuickCAN device.

Pin 9 (V++) is not connected at this time.

6.2. LED indication

The LED is lit when the QuickCAN device is connected to the PC. The lit LED indicates that the device is powered. Please note that the Windows device driver must be installed before the QuickCAN is ready to be used.

7. Device driver installation

7.1. For new purchase, skip to the step 6.2. For Interface Program **UPGRADE** from 1.0 to 1.1 or later, the following steps must be followed to ensure the QuickCAN device is working properly on the new device driver.

1) Detach the QuickCAN device from your PC USB port.
2) Remove the older version of the QuickCAN directory.
3) Install the new version of the Interface program.
4) Plug the QuickCAN device to your PC
5) Go to <Control panel> -> <system> -> <device manager> to remove the QuickCAN device driver. It should be under <Universal Serial bus controller>.
6) Detach and reattach the QuickCAN.
7) Windows should pop up the <Found new hardware Wizard> dialog.

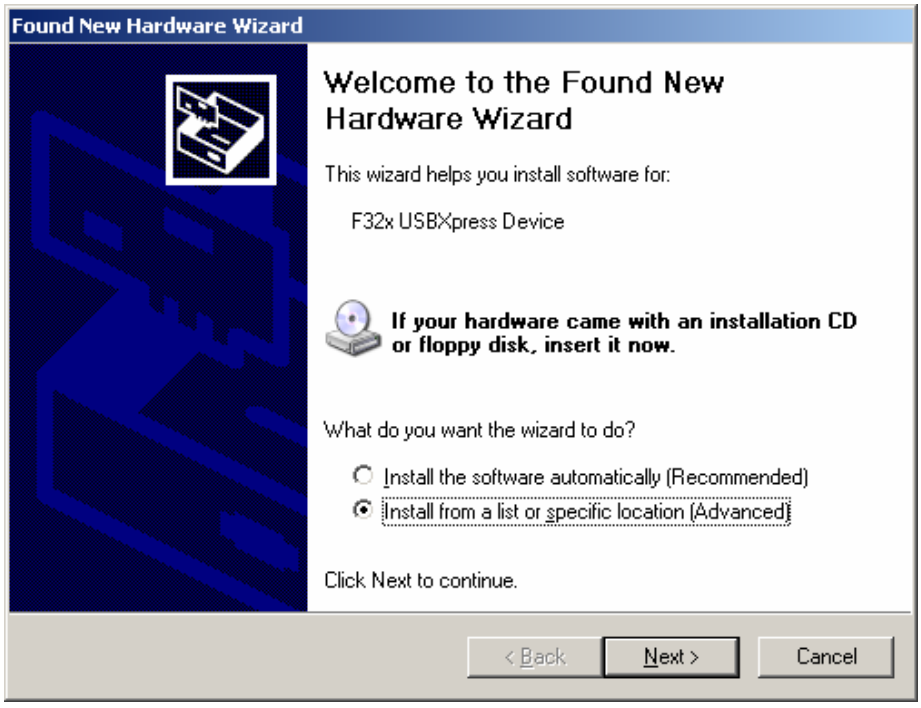
a)When Windows OS asks "Can Windows connect to Windows Update to search for software?", Click <this time only>.

b) Select "install from a list or specific location" and click <next>.

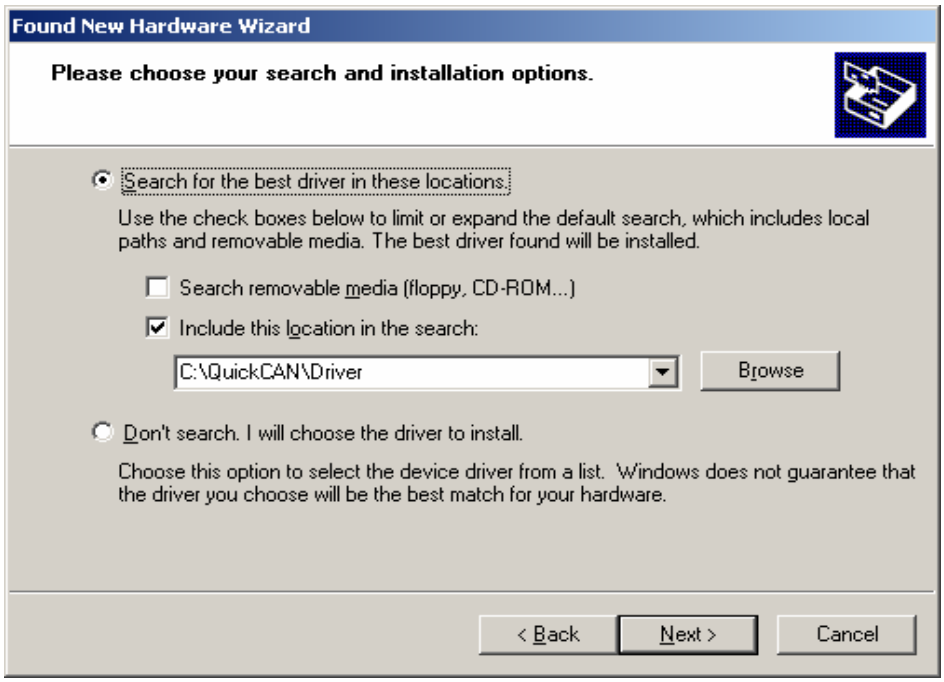
c) Check the "include this location in the search" box and browse to the driver directory that the driver directory that QuickCAN application is just installed.

d) Sometimes, Windows may ask the location of the file "SiF32x.sys". This file is also under the driver sub directory of the QuickCAN application.

7.2. The following steps are only applied to the new purchase user.
Plug in the QuickCAN device to your USB port through the USB A-B cable. The setup wizard should run automatically and a Welcome screen should appear as follows. Select option <Install from a list or specified location (Advanced)>. Then click the <Next> button.

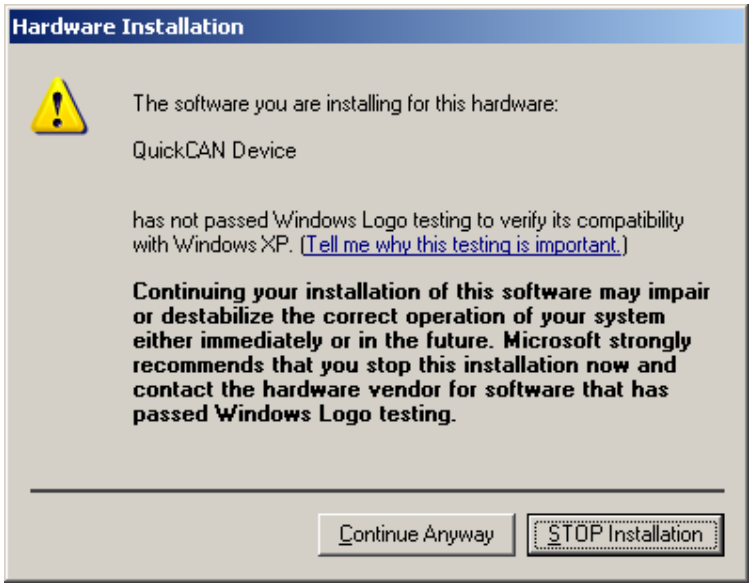

Found New Hardware Wizard
Welcome to the Found New Hardware Wizard
This wizard helps you install software for:
F32x USBXpress Device
If your hardware came with an installation CD or floppy disk, insert it now.
What do you want the wizard to do?
Install the software automatically (Recommended)
Install from a list or specific location (Advanced)
Click Next to continue.
< Back
Next >
Cancel

7.3. Select option <Search for the best driver in these locations>. Set the search location to the directory where QuickCAN driver is located, i.e. c:\QuickCAN\driver if you unzip QuickCAN.zip to c:\QuickCAN. Then click the <Next> button.


Found New Hardware Wizard
Please choose your search and installation options.
Search for the best driver in these locations.
Use the check boxes below to limit or expand the default search, which includes local paths and removable media. The best driver found will be installed.
Search removable media (floppy, CD-ROM...)
Include this location in the search:
C:\QuickCAN\Driver
Browse
Don't search. I will choose the driver to install.
Choose this option to select the device driver from a list. Windows does not guarantee that the driver you choose will be the best match for your hardware.
< Back
Next >
Cancel

7.4. Occasionally, you may get the following warning on some operating systems. This is because QuickCAN device driver has not been certified from Microsoft at this time. However, this installation

should not effect the operation of your system. Click the <Continues Anyway> button to continue installation.


Hardware Installation
The software you are installing for this hardware:
QuickCAN Device
has not passed Windows Logo testing to verify its compatibility with Windows XP. (Tell me why this testing is important.)
Continuing your installation of this software may impair or destabilize the correct operation of your system either immediately or in the future. Microsoft strongly recommends that you stop this installation now and contact the hardware vendor for software that has passed Windows Logo testing.
Continue Anyway
STOP Installation

7.5. Click the <Finish> button to complete the installation after the hardware wizard finishes installing QuickCAN device driver.

To verify the successfully installation of the driver, open Windows <control panel> → <System> → <Hardware> → <Device Manager>, the QuickCAN device shall be shown under “Universal Serial Bus controllers” tree.

8. CANmunicator

The CANmunicator (formally called QuickCAN interface application) provides users an easy and quick start to access Controller Area Network (CAN). The application uses the USB-2-CAN communication protocol to access the MCP2515 internal registers and initialize the proper MCP2515 registers to send and receive CAN messages.

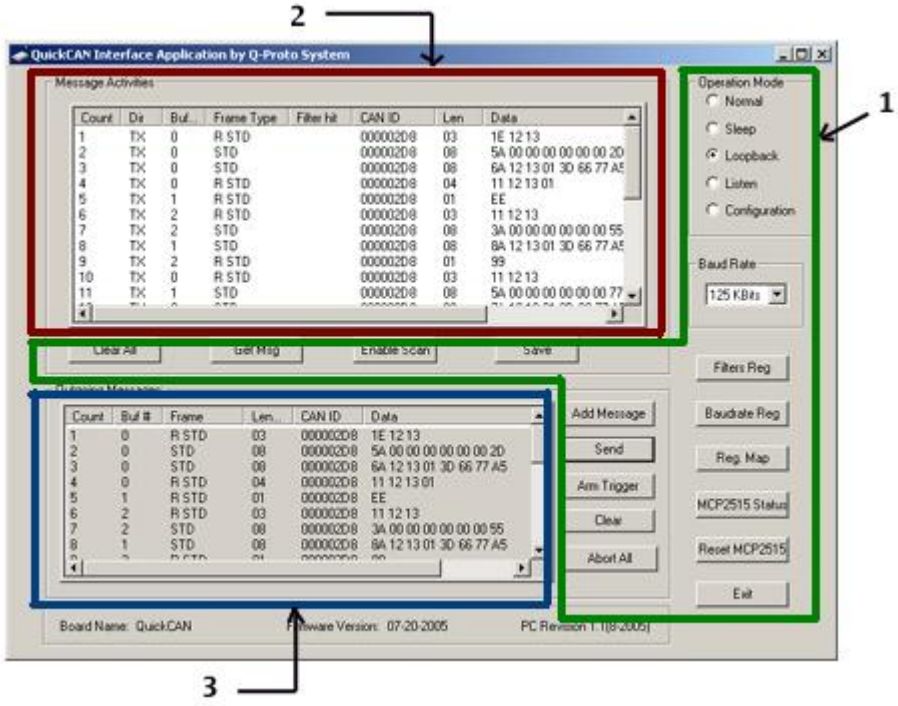
Overview:

The CANmunicator is composed of three major control groups.

1. Action buttons: Send command to access the internal registers of MCP2515, such as <Filter Registers>, <Baud rate registers> and <MCP2515 status> etc.

2. <Message Activities> pane: Displays both <Tx> and <Rx> activities across the CAN bus. Each message is time stamped by the Windows timer (not firmware timer) that is defined by the Timer interval in QuickCAN.ini.

3. <Outgoing message> pane: Lists all user-defined <Tx> data. Pressing either the button <Send> or <Arm Trigger> can send any messages in this list. <Outgoing message> list can be pre-loaded with data from a flat file. The data file name with full path is defined in QuickCAN.ini.



8.1. Clear All

This button removes all the activities in the <Message Activities> control list.

8.2. Get Message

This button gets all the received CAN message from the QuickCAN software FIFO. It cannot be used with <Enable Scan> simultaneously. It is disabled when <Enable Scan> is enabled.

The QuickCAN device uses Interrupt Method to receive CAN messages. User must make sure the receive bits in CANINTE are enabled before CAN messages can be received properly. Both RX1IE and RX0IE are enabled when the CANmunicator is launched. Users must re-enable those 2 bits after the <Reset MCP2515> is called before any message can be received. Other Interrupt bits shall be disabled (set to 0) in CANINTE.

8.3. Enable Scan

This button enables the automatic scanning of the incoming CAN message from QuickCAN MCP2515. Button <Get Msg> is disabled when <Enable Scan> is enabled.

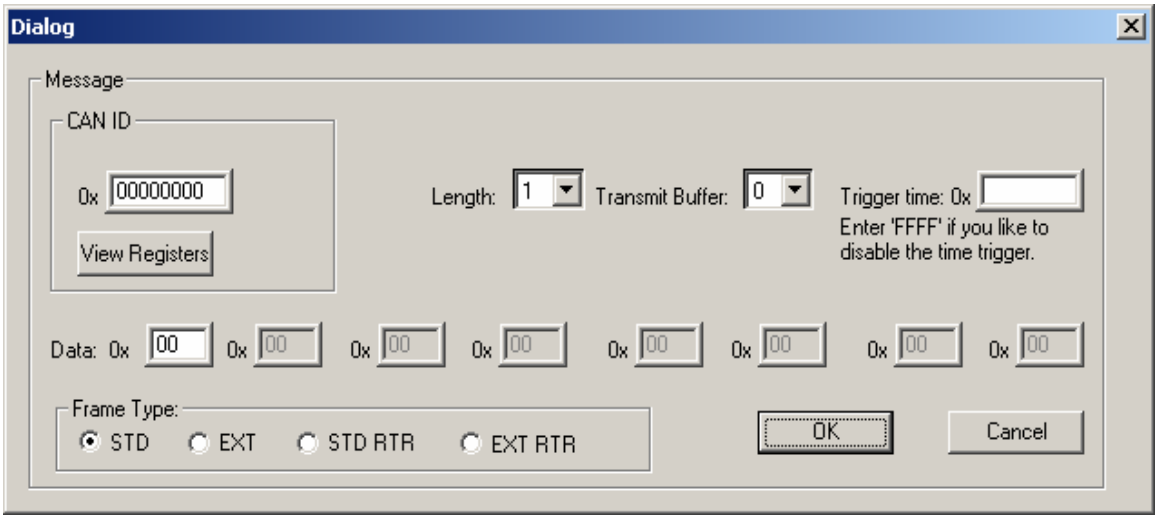
8.4. Save

This button saves all the data in <Message Activities> to a file in ASCII format. It is disabled when <Enable Scan> is enabled.

8.5. Add message

This button allows users to add a CAN message to the <outgoing message> list.

After <Add Message> button is pressed, a CAN dialog form pops up as following:



The <View registers> button allows user to see the generic data format of the CAN ID registers MCP2515.

8.6. Send

This button allows user to send the selected CAN message from the <Outgoing Messages> list.

8.7. Arm Trigger

This button sends all CAN messages that have the trigger time set in the <Outgoing Messages> list. These messages are called Time-Triggered messages. Once a time-triggered message is transmitted, its trigger time value is set to -1 to indicate the completion of transmission.

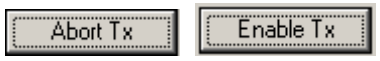
The trigger time and the QuickCAN application timer interval together determine when a time-triggered message is sent. The formula is shown as below.

Time to send the message = QuickCAN application timer value * trigger time value

Where: QuickCAN application timer interval value is set in the file QuickCAN.ini (See Chapter 9), and its default value is 10ms; the trigger time value set by users.

For example, the QuickCAN application timer value is 10 ms set in the file QuickCAN.ini, and the trigger is set to 0x64 (decimal 100), then this CAN message will be transmitted at 1000 ms after the button <Arm Trigger> is pressed.

8.8. Abort Tx ws Enable Tx

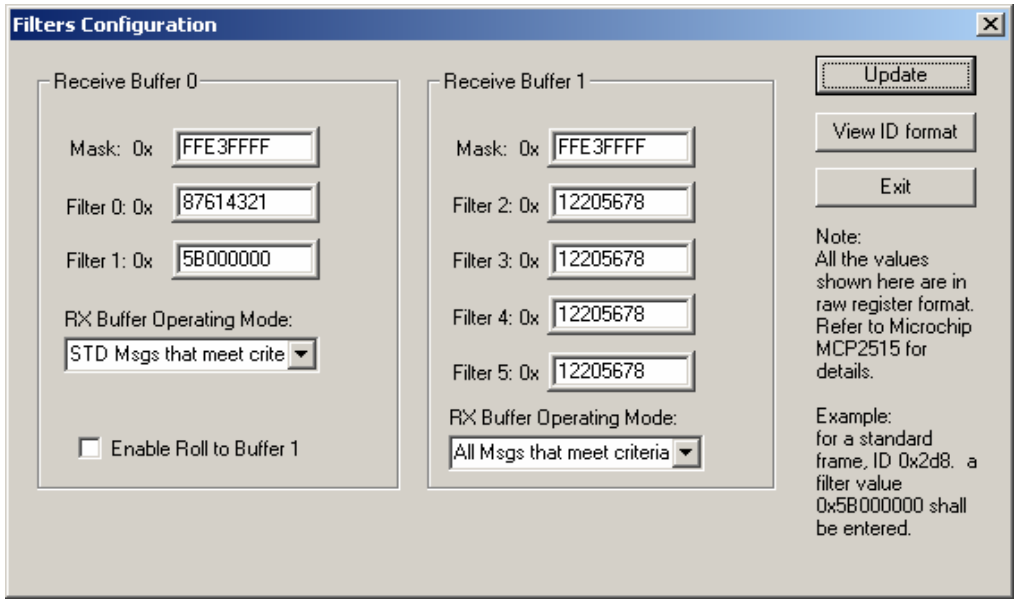


This button toggles the Global abort bit (ABAT bit) in the MCP2515. This bit can be manually set or cleared from <Reg.Map> dialog.

8.9. Filter Registers

This button allows users to edit the current values of MCP2515 reception filters and masks. It is disabled when button <Enable Scan> is enabled.

The values of MCP2515 reception filters and masks at application startup are defined in the file QuickCAN.ini, if any. Filters and masks changes made from the interface are NOT saved back to QuickCAN.ini.

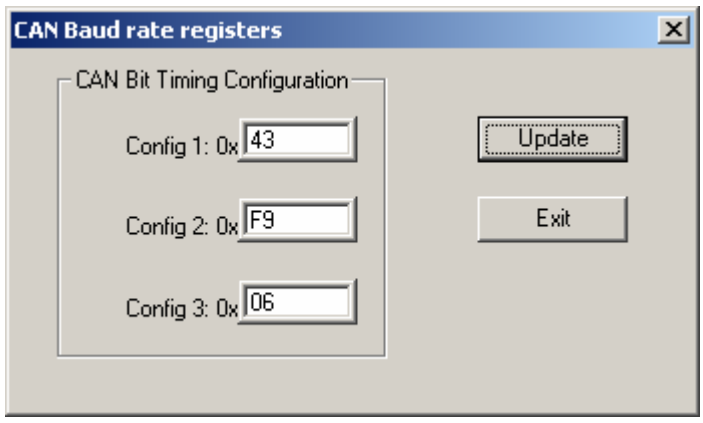

Filters Configuration
Receive Buffer 0
Mask: 0x FFE3FFFF
Filter 0: 0x 87614321
Filter 1: 0x 5B000000
RX Buffer Operating Mode:
STD Msgs that meet crite
Enable Roll to Buffer 1
Receive Buffer 1
Mask: 0x FFE3FFFF
Filter 2: 0x 12205678
Filter 3: 0x 12205678
Filter 4: 0x 12205678
Filter 5: 0x 12205678
RX Buffer Operating Mode:
All Msgs that meet criteria
Update
View ID format
Exit
Note:
All the values shown here are in raw register format. Refer to Microchip MCP2515 for details.
Example:
for a standard frame, ID 0x2d8. a filter value 0x5B000000 shall be entered.

<View ID format> allows user to view the filter register into converted ID format. For example, a standard frame, filter register value 0x5B000000 is shown as 0x000002D8 after the conversion.

8.10. Baud rate Register

This button allows users to change the baud rate registers for the customized CAN speed. It is disabled when button <Enable Scan> is enabled.

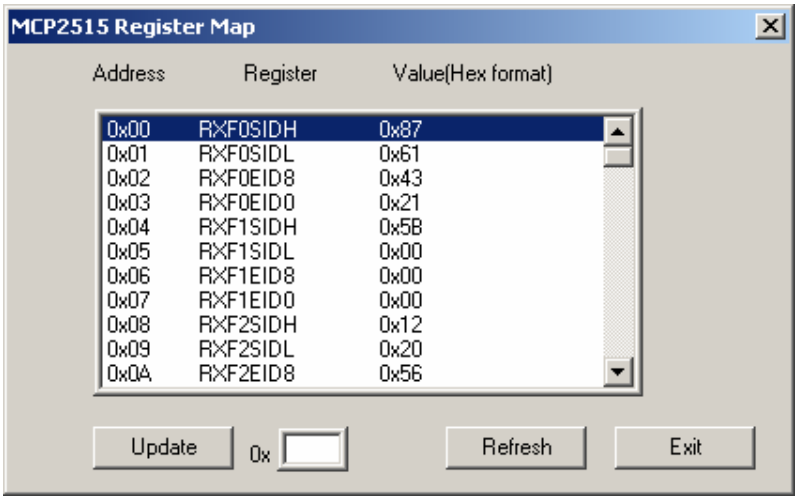
The MCP2515 chip is running at 16MHz.



8.11. Register map

This button allows users to retrieve and edit the values of all the MCP2515 internal registers except for baud rate and filter registers. Refer to MCP2515 datasheet for details.

Note: The MCP2515 operation mode is NOT changed during this operation. Hence, the values of the baud rate registers and filter registers may not be shown properly as they can only be updated in configuration mode.

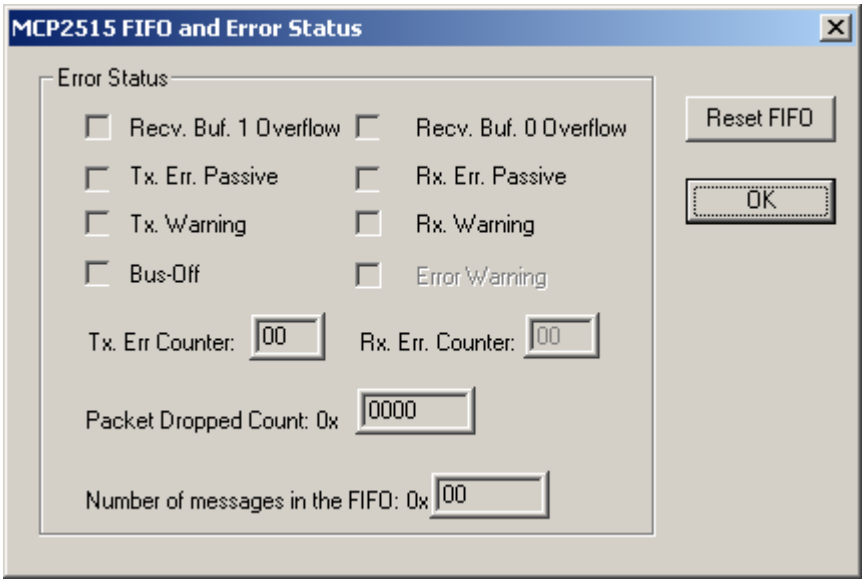


8.12. MCP2515 Status

This button shows the error status of CAN network. Refer to MCP2515 datasheet for details. It also shows the status of the receiving FIFO.

- Packet Dropped count: the number of dropped packet when the receiving FIFO is full. The QuickCAN device has 32 software FIFOs to receive CAN messages.

- Number of messages in the FIFO: the number of received CAN messages in the software FIFO.



8.13. Reset MCP2515

This button resets the MCP2515 to its power up state.

8.14. Exit

This button exits the QuickCAN application.

8.15. Operation mode

This button sets the MCP2515 operation mode.

8.16. Baud Rate

This dropdown list allows the users to set the MCP2515 to a pre-configured baud rate.

9. USB-2-CAN Communication protocol

The QuickCAN device provides a set of communication protocol for users to develop their own applications. The protocol is developed with the Silicon Laboratories USBXpress driver development kits, and allows users to easily create USB application to the QuickCAN device.

It is strongly recommended that USBXpress be used to develop applications because of its easy-to-use and compatibility with the QuickCAN device. For details of the USBXpress, please refer to USBXpress programmer guide from http://www.cygnal.org/.

QuickCAN application version	USBXpress version used.
1.0	USBXpress 1.4
1.1	USBXpress 2.1

The communication protocol is asynchronous, which means one command and one response. No command shall be issued before the previous response is received.

- All commands start with '@' followed by QuickCAN command byte and ended with checksum.
- All responses start with '@' followed by QuickCAN command byte plus returned data, if any, and the checksum, except for the response to 'J' command. 'J' response does not have checksum appended.
- If any error occurs during the operation, the first two bytes of the response are '@E' followed by the error code and checksum.
- The user application shall check for error status from the response before proceeding to the next operation.

The XOR checksum method is applied to all the commands and responses **except for the response to 'J' command**. For example, if a request 0x40 0x41 is sent to the device, a checksum byte 0x01 shall be appended to the original data. Therefore the real time outgoing data is 0x40 0x41 0x01; the same XOR method applies to responses except for the response to 'J' command.

In the USB-2-CAN protocol, QuickCAN command byte refers to the command sent to the QuickCAN device. Usually, this command byte is the second byte the command packet. The MCP2515 command byte refers to the command sent to MCP2515, which could be embedded in the whole packet.

The USB-2-CAN protocol is an extension of the MCP2515 SPI command set. It allows developers to interface with MCP2515 through the USB interface. For detailed of the MCP2515 SPI command set, see MCP2515 datasheet, SPI interface section (section 12).

In the USB-2-CAN protocol, even the length of the response is always 16 bytes with the exception of the 'J' response (multiple of 16 bytes), only first few bytes are meaningful. This is because the QuickCAN firmware software buffer is rounded at 16 bytes. The following section depicts the details of the commands set.

9.1. Command R

Description: Read register

Format:

Byte 0	Byte 1	Byte 2	Byte 3
@	R	Register address	Checksum

Response:

Byte 0	Byte 1	Byte 2	Byte 3
@	R	Register value	Checksum

Error response:

Byte 0	Byte 1	Byte 2	Byte 3
@	E	Error code	Checksum

Error code:

1	MCP2515 Error
2	Invalid QuickCAN command
3	Invalid checksum

9.2. Command W

Description: Update register

Format:

Byte 0	Byte 1	Byte 2	Byte 3	Byte 4
@	W	Register address	Value	Checksum

Response:

Byte 0	Byte 1	Byte 2
@	W	Checksum

Error response:

Byte 0	Byte 1	Byte 2	Byte 3
@	E	Error code	Checksum

Error code:

1	MCP2515 Error
2	Invalid QuickCAN command
3	Invalid checksum

9.3. Command C:

Description: Read RX buffer

Format:

Byte 0	Byte 1	Byte 2	Byte 3
@	C	MCP2515 Command byte	Checksum

MCP2515 Command byte (refer to MCP2515 datasheet):

- 0: RX buffer starts from RXB0SIDH.

- 1: RX buffer starts from RXBD0
- 2: RX buffer starts from RXB1SIDH.
- 3: RX buffer starts from RXBD1

Response:

Byte 0	Byte 1	Byte 2 ~ 9 or 2 ~ 14	Byte 10 or 15
@	C	Data	Checksum

Error response:

Byte 0	Byte 1	Byte 2	Byte 3
@	E	Error code	Checksum

Error code:

1	MCP2515 Error
2	Invalid QuickCAN command (byte 1)
3	Checksum
4	Invalid MCP2515 Command byte (byte 2)

9.4. Command X

Description: Load TX buffer

Format:

Byte 0	Byte 1	Byte 2	Byte 3 ~ 15	Byte 16
@	X	MCP2515 Command byte	Data (see MCP2515 datasheet)	Checksum

MCP2515 Command byte:

0x40: TX buffer 0, start at TXB0SIDH
0x41: TX buffer 0, start at TXB0D0
0x42: TX buffer 1, start at TXB1SIDH
0x43: TX buffer 1, start at TXB1D0
0x44: TX buffer 2, start at TXB2SIDH
0x45: TX buffer 2, start at TXB2D0

Response:

Byte 0	Byte 1	Byte 2
@	X	Checksum

Error response:

Byte 0	Byte 1	Byte 2	Byte 3
@	E	Error code	Checksum

Error code:

1	MCP2515 Error

2	Invalid QuickCAN command
3	Invalid Checksum

9.5. Command I

Description: Get Rx status

This command is intended for QuickCAN firmware use even it is listed here. Developers should use the Status byte in the 'J' command to determine the RX buffer number and matched filter.

The ISR (interrupt service routine) in the QuickCAN firmware is active all the time. It copies the RxStatus to the software receiver FIFO along with the data when message arrives, so it can be retrieved later with the 'J' command. The RxStatus returned from the 'I' command may not be real time because this value is refreshed each time a new message arrives.

Format:

Byte 0	Byte 1	Byte 2
@	I	Checksum

Response:

Byte 0	Byte 1	Byte 2	Byte 3
@	I	Status value (see MCP2515 <SPI interface> for details)	Checksum

Error response:

Byte 0	Byte 1	Byte 2	Byte 3
@	E	Error code	Checksum

Error code:

1	MCP2515 Error
2	Invalid QuickCAN command
3	Invalid Checksum

9.6. Command A:

Description: Reset MCP2515

Format:

Byte 0	Byte 1	Byte 2
@	A	Checksum

Response:

Byte 0	Byte 1	Byte 2
@	A	Checksum

Error response:

Byte 0	Byte 1	Byte 2	Byte 3
@	E	Error code	Checksum

Error code:

1	MCP2515 Error
2	Invalid command
3	Invalid Checksum

9.7. Command T:

Description: Set RTS command

Format:

Byte 0	Byte 1	Byte 2	Byte 3
@	T	Command byte	Checksum

Command byte (refer to MCP2515 datasheet):

- 0: None.
- 1: TX0
- 2: TX1
- 3: TX0 & TX1
- 4: TX2
- 5: TX2 & TX0
- 6: TX2 & TX1
- 7: TX2, TX1, TX0

Response:

Byte 0	Byte 1	Byte 2
@	T	Checksum

Error response:

Byte 0	Byte 1	Byte 2	Byte 3
@	E	Error code	Checksum

Error code:

1	MCP2515 Error
2	Invalid QuickCAN command
3	Invalid Checksum
4	Invalid MCP2515 byte (byte 2)

9.8. Command M:

Description: Send BIT modify command

Format:

Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5
@	M	Address	Mask	Data	Checksum

Response:

Byte 0	Byte 1	Byte 2
@	M	Checksum

Error response:

Byte 0	Byte 1	Byte 2	Byte 3
@	E	Error code	Checksum

Error code:

1	MCP2515 Error
2	Invalid QuickCAN command
3	Invalid Checksum

9.9. Command D:

Description: Get multiple bytes from MCP2515 (up to 8 bytes each time)

Format:

Byte 0	Byte 1	Byte 2	Byte 3	Byte 4
@	D	Number of byte to retrieve	Starting address	Checksum

Response:

Byte 0	Byte 1	Byte 2	Byte 3 ~ 10	Byte 4 or 5 or …11
@	D	Number of retrieved byte	Data	Checksum

Error response:

Byte 0	Byte 1	Byte 2	Byte 3
@	E	Error code	Checksum

Error code:

1	MCP2515 Error
2	Invalid QuickCAN command
3	Invalid Checksum
4	Byte 2 > 8

9.10. Command Q:

Description: Write multiple bytes to MCP2515 (up to 8 bytes each time)

Format:

Byte 0	Byte 1	Byte 2	Byte 3	Byte 4 ~ 11	Byte 5 or 6 or … 12

@	Q	Number of byte to write	Starting address	Data	Checksum

Response:

Byte 0	Byte 1	Byte 3
@	Q	Checksum

Error response:

Byte 0	Byte 1	Byte 2	Byte 3
@	E	Error code	Checksum

Error code:

1	MCP2515 Error
2	Invalid QuickCAN command
3	Invalid Checksum
4	Byte 2 > 8

9.11. Command V:

Description: Get the error status, packet dropped count from the MCP 2515

Format:

Byte 0	Byte 1	Byte 2
@	V	Checksum

Response:

Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
@	V	TEC	REC	EFLG	Dropped packet count high byte	Dropped packet count low byte	Number of CAN Msg in the FIFO	Checksum

Error response:

Byte 0	Byte 1	Byte 2	Byte 3
@	E	Error code	Checksum

Error code:

1	MCP2515 Error
2	Invalid QuickCAN command
3	Invalid Checksum

9.12. Command Z:

Description: Reset the MCP 2515 RX FIFO. The number of CAN messages in FIFO is also reset to zero.

Format:

Byte 0	Byte 1	Byte 2
@	Z	Checksum

Response:

Byte 0	Byte 1	Byte 2
@	Z	Checksum

Error response:

Byte 0	Byte 1	Byte 2	Byte 3
@	E	Error code	Checksum

Error code:

1	MCP2515 Error
2	Invalid QuickCAN command
3	Invalid Checksum

9.13. Command J:

Description: Request the MCP2515 to send received messages up to 32 packets to the host.

Format:

Byte 0	Byte 1	Byte 2
@	J	Checksum

Response:

Transfers up to 32 packets to the host. Note that this response has no checksum byte. The format of the packet is shown below:

Byte 0	Byte 1	Byte 2	Byte 3-6	Byte 7	Byte 8 ~ 15	Byte 16
@	J	Rx Status	CAN ID	Length	Data	N/A
Byte 17	Byte 18	Byte 19	Byte 20-23	Byte 24	Byte 25~32	Byte 33
@	J	Rx Status	CAN ID	Length	Data	N/A
Byte 34	Byte 35	Byte 36	Byte 37-40	Byte 41	Byte 42~49	Byte 50
@	J	Rx Status	CAN ID	Length	Data	N/A
. . . Up to 32 packets						

Rx Status:

Bit 7	Bit 6	Received message
0	0	No Rx
0	1	Message in RxB0

1	0	Message in RxB1
1	1	Messages in both buffers

Frame type shall be checked from the message packet

Bit 2	Bit 1	Bit 0	Filter match
0	0	0	RXF0
0	0	1	RXF1
0	1	0	RXF2
0	1	1	RXF3
1	0	0	RXF4
1	0	1	RXF5
1	1	0	RXF0 (roll over to RxB1)
1	1	1	RXF1 (roll over to RxB1)

Error response:

Byte 0	Byte 1	Byte 2	Byte 3
@	E	Error code	Checksum

Error code:

1	MCP2515 Error
2	Invalid QuickCAN command
3	Invalid Checksum
5	No data (actually not an error)

10. Initialization

Configuration file QuickCAN.ini is used to initialize the MCP2515 when the CANmunicator is launched. This file is located at the same directory of the QuickCAN.exe. The following parameters can be set in the QuickCAN.ini file.

10.1. Initialization parameters

- SWITCH

  Format:
  ON=1 or 0

  When ON=1, QuickCAN.ini file is used to initialize the MCP2515
  When ON=0, QuickCAN.ini file is not used at all.

- TIMER

  Format:
  INTERVAL=<value>

  Set up the CANmunicator timer interval. This interval and the trigger time value together determine when to send a Time-Triggered CAN message from the <outgoing messages> list.

  User shall carefully set the timer interval that is sufficient to process all the timer-triggered messages in the <Outgoing Messages> list control.

- CONFIG

  Format:
  MASK0 = <value>
  MASK1 = <value>
  FILTER0 = <value>
  FILTER1 = <value>
  FILTER2 = <value>
  FILTER3 = <value>
  FILTER4 = <value>
  FILTER5 = <value>
  CNF3 = <value>
  CNF2 = <value>
  CNF1 = <value>
  ACCEPTANCE0 = <value>
  ACCEPTANCE1 = <value>

  Where MASK0, MASK1, FILTER0, FILTER1, FILTER2, FILTER3, FILTER4, FILTER5, CNF3, CNF2, CNF1 are the mask, filter and baud rate registers of the MCP2515.

  ACCEPTANCE parameter defines the type of CAN message that can be received in filter 0 or 1 respectively. See table below for detail.

ACCEPTANCE value	Message to receive
0	Filter is off
1	Receive only valid messages with extended identifiers that meet filter criteria

2	Receive only valid messages with standard identifiers that meet filter criteria
3	Receive all valid messages using either standard or extended identifiers that meet the filter criteria

- MAX_LIST_COUNT

  Format:
  COUNT=<value>

  The <value> is the maximum number of messages that can be displayed in the <Message Activities> list control. When this limit is reached, the first come first out method is used. Setting this limit to zero allows <Message Activities> control to continuously display all the traffic without flushing.

  Time to flush the <Message Activities> may vary on different OS and CPU.

- FILE

  Format:
  FILE=<file name with full path>

  The <name> is file name of the ASCII file that has all the TX data to be loaded into the <Outgoing Messages> list.

  Format of the file:

  1. Packet format

     <TxBuf Id>, <CAN id>, <Frame type>, <length>, <data>, <triggered time>,

     Each packet ends with ',' followed by the carriage return.

  2. <Frame type> must be set to one of the following values:

     (1) STD
     (2) EXT
     (3) R STD
     (4) R EXT

  3. A line starting with character '#' is considered as comment.

  4. When <triggered time> is -1, the packet will not be transmitted when <Arm Trigger> is active.

10.2. Examples

10.2.1. QuickCAN.ini

```
[SWITCH]
ON=1

[TIMER]
```

INTERVAL=30 ← timer interval is 30 ms. This value is used by TX trigger event.

[CONFIG]
MASK0=0x0
MASK1=0xFFFFFFFF
FILTER0=0x87654321
FILTER1=0x5B000000
FILTER2=0x12345678
FILTER3=0x12345678
FILTER4=0x12345678
FILTER5=0x12345678
CNF3=0x6
CNF2=0xf9
CNF1=0x43
;0: filter off, 1: standard 2: extended 3: standard or extended
ACCEPTANCE0=1 ← accept standard frames that meet the filter criteria
ACCEPTANCE1=3 ← accept both standard and extended frames,

[FILE]
NAME=c:\quickcan\host\debug\init.txt

10.2.2. Example of the init.txt specified in QuickCAN.ini [FILE] option.

#TxBuf Id, CAN id, Frame type, length, data, triggered time, ← This is a comment line

0,0x2d8,R STD,0x3,0x1E,0x12,0x13,-1, ←remote standard frame, id is 0x2d8, length is 0x3 and no time trigger

0,0x2d8,STD,0x8,0x5a,0,0,0,0,0,0,0x2d,0x567, ← standard frame, id is 0x2d8, length is 8, data packets are 0x5a, 0, 0, 0, 0, 0, 0, 0x2d, triggered time is 0x567 count of the preset application timer interval.

[MAX_LIST_CNT]
COUNT=10000 ← <Message Activities> will display as many as 10000 messages before flushing.

11. Revision History

Document Revision	Software Revision	Date	Comment
1.0	1.0	11-05-2004	Initial release
1.1	1.1	7-22-2005	1) Add section 6.1 to show how to upgrade Interface program from 1.0 to 1.1. 2) Add description for <view register> button under <Add message> dialog. 3) The interface program is USBXpress.2.1 compatible. 4) Add <view ID format> in filter dialog to show the converted ID.

			format from registers values.
1.2	1.2	8-18-2005	The QuickCAN interface program is now called CANmunicator. Fixed the description in Section 10. Init file should now be in the same directory of quickcan.exe file instead of c:\windows <Message activities> pane will not reset the content and first in first out method is used. <Abort All> button is changed to <Abort Tx> and <Enable Tx> to toggle the ABAT bit. Users are no longer need to clear this bit manually to re-enable the transmission.
1.3	1.3	11-1-2005	Add <refresh> button in the Registers map dialog.